

www.bijishequ.com

搜索你想要的内容

搜索

关注微信公众号: PMvideo

【Spring学习36】Spring事务(6): 声明式事务(集合Mybatis)

作者: soonfly (/authorarticle.html?author=soonfly) 2017-04-20 ☆ 收录到我的专题 (/select.html?articleId=401736)

标签 [schema](http://www.bijishequ.com/info/search.html?searchText=schema) (<http://www.bijishequ.com/info/search.html?searchText=schema>) [wwwspringframeworkorg](http://www.bijishequ.com/info/search.html?searchText=wwwspringframeworkorg) (<http://www.bijishequ.com/info/search.html?searchText=wwwspringframeworkorg>) [Spring](http://www.bijishequ.com/info/search.html?searchText=Spring) (<http://www.bijishequ.com/info/search.html?searchText=Spring>) [http](http://www.bijishequ.com/info/search.html?searchText=http) (<http://www.bijishequ.com/info/search.html?searchText=http>) [事务](http://www.bijishequ.com/info/search.html?searchText=事务) (<http://www.bijishequ.com/info/search.html?searchText=事务>)

Spring声明式事务实现过程中,在配置文件中关于事务配置总是由三个组成部分,分别是DataSource、TransactionManager和代理机制这三部分,无论哪种配置方式,一般变化的只是代理机制这部分。DataSource、TransactionManager这两部分只是会根据数据访问方式有所变化。

比如使用Hibernate进行数据访问时,DataSource实际为SessionFactory,TransactionManager的实现为HibernateTransactionManager。使用myBatis时用的是JDBC事务管理器,因此TransactionManager的实现为DataSourceTransactionManager。

先假设我们在myBatis中自己写事务,是这个样子:

```
1 public class StudentService {
2     public Student createStudent(Student student) {
3         SqlSession sqlSession = MyBatisUtil.getSqlSessionFactory()
4             .openSession();
5         try {
6             StudentMapper mapper = sqlSession.getMapper(StudentMapper.class);
7             mapper.insertAddress(student.getAddress());
8             mapper.insertStudent(student);
9             sqlSession.commit();
10            return student;
11        } catch (Exception e) {
12            sqlSession.rollback();
13            throw new RuntimeException(e);
14        } finally {
15            sqlSession.close();
16        }
17    }
18 }
```

按这个方式,在每一个需要用到事务的方法中,添加事务的提交、回滚、关闭等。

现在我们来使用spring的事务处理能力。首先在Spring的配置文件中配置TransactionManager。

Spring全局配置beans.xml:

```

1 <?xml version="1.0" encoding="UTF-8" ?>
2 <beans xmlns="http://www.springframework.org/schema/beans" xmlns:context="http://www.springframework.org/schema/cont
3 xt" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:c="http://www.springframework.org/schema/c" xmlns:tx=
4 "http://www.springframework.org/schema/tx" xmlns:aop="http://www.springframework.org/schema/aop" xsi:schemaLocation=
5 "http://www.springframework.org/schema/beans http://www.springframework.org/schema/beans/spring-beans.xsd http://www.
6 springframework.org/schema/context http://www.springframework.org/schema/context/spring-context-4.2.xsd http://www.sp
7 ringframework.org/schema/tx http://www.springframework.org/schema/tx/spring-tx-4.2.xsd http://www.springframework.or
8 g/schema/aop http://www.springframework.org/schema/aop/spring-aop-4.2.xsd">
9
10 <bean id="dataSource" class="org.springframework.jdbc.datasource.DriverManagerDataSource">
11 <property name="driverClassName" value="${jdbc.driverClassName}"></property>
12 <property name="url" value="${jdbc.url}"></property>
13 <property name="username" value="${jdbc.username}"></property>
14 <property name="password" value="${jdbc.password}"></property>
15 </bean>
16
17 <bean id="sqlSessionFactory" class="org.mybatis.spring.SqlSessionFactoryBean">
18 <property name="dataSource" ref="dataSource" />
19 <property name="typeAliases" value="com.owen.mybatis.domain.Student" />
20 <property name="typeHandlers" value="com.owen.mybatis.typehandlers.PhoneTypeHandler" />
21 <property name="typeHandlersPackage" value="com.owen.mybatis.typehandlers" />
22 <property name="mapperLocations" value="classpath*:com/mybatis3/**/*.xml" />
23 </bean>
24
25 <bean id="sqlSession" class="org.mybatis.spring.SqlSessionTemplate">
26 <constructor-arg index="0" ref="sqlSessionFactory" />
27 </bean>
28
29 <bean class="org.mybatis.spring.mapper.MapperScannerConfigurer">
30 <property name="basePackage" value="com.owen.mybatis.mappers" />
31 </bean>
32
33 <!-- =====事务配置===== -->
34 <!-- 基于注解的事务处理特性, Spring需要先使用下面的配置 -->
35 <tx:annotation-driven transaction-manager="transactionManager" />
36
37 <!-- 事务管理器 -->
38 <bean id="transactionManager" class="org.springframework.jdbc.datasource.DataSourceTransactionManager">
39 <property name="dataSource" ref="dataSource" />
40 </bean>
41 </beans>

```

现在可以在Spring的服务的Bean中注解@ Transactional。这个注解表明每个方法都是Spring来管理的。如果方法成功处理,那么Spring就会提交事务;如果就去处理过程出现了错误,那么事务就会被回滚。当然, Spring将会关心MyBatis的转换过程是否出现Exceptions的DataAccessExceptions的异常栈。

在DAO上需加上@Transactional注解,如下:

```

1 package twm.spring.transactiondemo.dao;
2
3 import java.util.List;
4 import org.hibernate.SessionFactory;
5 import org.springframework.beans.factory.annotation.Autowired;
6 import org.springframework.orm.hibernate3.support.HibernateDaoSupport;
7 import org.springframework.stereotype.Component;
8 import twm.spring.transactiondemo.pojo.User;
9
10 @Transactional(propagation = Propagation.REQUIRES_NEW, isolation = Isolation.READ_COMMITTED, noRollbackFor = { TExcep
11 tion.class }, readOnly = true, timeout = 3)
12 @Component
13 public class StudentService {
14 @Autowired
15 private StudentMapper studentMapper;
16
17 public Student createStudent(Student student) {
18 studentMapper.insertAddress(student.getAddress());
19 if(student.getName().equalsIgnoreCase("")) {
20 throw new RuntimeException("Student name should not be empty.");
21 }
22 studentMapper.insertStudent(student);
23 return student;
24 }
25 }

```

另附参考文章:
根据代理机制的不同,总结了五种Spring事务的配置方式,见
<http://www.blogjava.net/robbie/archive/2009/04/05/264003.html> (<http://www.blogjava.net/robbie/archive/2009/04/05/264003.html>)



登录

(/login.htm

l?redirectU

rl=%2Fdetail/401736)

0条评论或问题

社区邀请

笔记社区是一个面向中高端IT开发者、程序员的知识共享社区,通过网络抓取与文章分类总结,由专家为用户提供高质量的专题文章系列。
邀请您成为社区专家 >> (<http://www.bijishequ.com/creat.html>)
原文链接: <http://blog.csdn.net/soonfly/article/details/70305717>
声明: 所有文章资源均从网络抓取,如果侵犯到您的著作权,请联系删除文章。联系方式请关注微信公众号PMvideo【锤子视频-程序员喜欢的短视频】,或者加笔记社区开发者交流群 628286713。

4 签到

January 今日签到1人

相关标签

Struts (/tag/list?tagId=1440)

loc (/tag/list?tagId=1441)

Spring (/tag/list?tagId=1438)

SpringMVC (/tag/list?tagId=1560)

ibatis (/tag/list?tagId=1987)

Hibernate (/tag/list?tagId=1986)

Mybatis (/tag/list?tagId=1985)

Oracle (/tag/list?tagId=1747)

sql (/tag/list?tagId=1748)

SQLite (/tag/list?tagId=1749)

Mysql (/tag/list?tagId=1746)

Web服务 (/tag/list?tagId=1619)

ATOM (/tag/list?tagId=1623)

Xml (/tag/list?tagId=1621)

IWorkbench (/tag/list?tagId=519)

EnumMap (/tag/list?tagId=518)

HttpRequest (/tag/list?tagId=516)

StringUtils (/tag/list?tagId=515)

RSS (/tag/list?tagId=1622)

SOAP (/tag/list?tagId=1620)

JDK动态代理 (/tag/list?tagId=1444)

依赖注入 (/tag/list?tagId=1442)

AOP (/tag/list?tagId=1443)

CGLib (/tag/list?tagId=1445)

架构 (/tag/list?tagId=1977)

持久层 (/tag/list?tagId=1983)

ORM框架 (/tag/list?tagId=1984)

缓存 (/tag/list?tagId=1988)

关系型数据库 (/tag/list?tagId=1740)

OceanBase (/tag/list?tagId=1741)

分布式数据库 (/tag/list?tagId=1739)

数据存储 (/tag/list?tagId=1729)

J2EE (/tag/list?tagId=2397)

web应用开发 (/tag/list?tagId=1595)

web应用 (/tag/list?tagId=1593)

we框架 (/tag/list?tagId=1594)

编程语言 (/tag/list?tagId=2761)

java (/tag/list?tagId=2396)

相关文章

- Springboot 和 Mybatis集成开发 (/detail/576721?p=)
- mybatis generator插件编写 (/detail/576006?p=)
- 概率算法 (/detail/574875?p=)
- ActFramework 发布 1.6.0 正式版以贺新年 (/detail/577630?p=)
- 扩展mybatis-generator自动生成代码项目 (/detail/576158?p=)
- 有趣的MyBatis——延迟加载 (/detail/576297?p=)
- 有趣的MyBatis——延迟加载 (/detail/574718?p=)
- Mybatis-Plus——为Mybatis镀一层金 (/detail/576930?p=)
- mybatis分页插件报错 (/detail/576580?p=)
- java 视频播放 多人及时弹幕技术 代码生成器 websocket springmvc mybatis SSM (/detail/576244?p=)

相关专题



Mybatis使用和常见问题

本专题主要集中在Mybatis使用和常见问题上,收集平时开发中遇到的mybatis相关问题以及解决方案,便于回顾查看。

共 4 篇文章

(/subject/123)



Mybatis学习笔记

Mybatis执行CachingExecutor、Mybatis源码之Statement处理器CallableStatementHandler、Mybatis主键生成器SelectKeyGenerator

共 15 篇文章

(/subject/128)



Mybatis简单自学系列

学习框架最好还是以使用为主，使用的不熟，一味钻研原理也是徒劳，所以这里决定写个最简单的实例，尽量多包括一些知识点。

共 5 篇文章

(/subject/231)



Hibernate入门基础

Hibernate是一个著名的对象关系映射框架，使得Java程序员可以随心所欲的使用对象编程思维来操纵数据库。专题是Hibernate的入门总结，主要介绍ORM和hibernate的概念，如何在eclipse中创建hibernate实例及相关配置文件的详解说明。

共 18 篇文章

(/subject/51)



Mybatis使用和常见问题

本专题主要集中在Mybatis使用和常见问题上，收集平时开发中遇到的mybatis相关问题以及解决方案，便于回顾查看。

共 4 篇文章

(/subject/123)



Mybatis学习笔记

Mybatis执行CachingExecutor、Mybatis源码之Statement处理器CallableStatementHandler、Mybatis主键生成器SelectKeyGenerator

共 15 篇文章

(/subject/128)



Mybatis简单自学系列

学习框架最好还是以使用为主，使用的不熟，一味钻研原理也是徒劳，所以这里决定写个最简单的实例，尽量多包括一些知识点。

共 5 篇文章

(/subject/231)



并发编程之并发队列

队列在并发编程中有很多的应用场景，是并发编程容器中的重要组成部分，本专题收录了一些比较经典的并发队列文章。

共 5 篇文章

(/subject/162)



JAVA多线程专题

所谓并发编程是指在一台处理器上“同时”处理多个任务。并发是在同一实体上的多个事件。

共 0 篇文章

(/subject/348)



RocketMQ实战